Human Character-oriented Animated GIF Generation Framework

Ghulam Mujtaba
Department of Computer Education
Sungkyunkwan University
Seoul 03063, South Korea
gmujtabakorai@gmail.com

Eun-Seok Ryu
Department of Computer Education
Sungkyunkwan University
Seoul 03063, South Korea
esryu@skku.edu

***Abstract*—Click-through rate (CTR) is a critical metric to boost the popularity of newly published videos on streaming platforms. Humans and human-like characters play a significant role in GIF selection and improving the CTR of the video. This paper proposes a new lightweight method to generate human character-oriented animated GIFs using the end-user device's computational capabilities. Instead of analyzing full video, the proposed method analyzes the lightweight thumbnail containers to decrease computational complexity in the GIF generation process. Moreover, it uses the segment to generate the GIF and reduced valuable network bandwidth and storage demands in the user end. A feed-forward 2D deep neural network trained on the CelebA dataset is designed to detect humans or human-like characters and their gender. Experimental evaluations and results performed in 10 full videos showed that the proposed method is 2.34 times more computationally efficient than the SoA approach. The proposed method is designed to support end-user devices with different computational capabilities.**

***Index Terms*—animated GIF, human character, client-driven, video analysis.**

I. Introduction

WITH access to high-speed internet and usage of social media website, users have access to the vast amount of multimedia content. Streaming platforms use recommended algorithms and animated Graphical Interchange Format (GIF) images to reduce this information overload. The recommended algorithms aim to provide suggestions for the most relevant items depending on the interests/activity of the user. Meanwhile, the GIF displays an instant preview of the recommended video. Compared to the full video, the GIF is shorter in duration (3–15 seconds) and lighter in weight [1]. Thus, the GIF requires less bandwidth and a lower time lag to retrieve. This makes GIFs irreplaceable on streaming platforms. However, they are being generated and presented without the consent of the user.

Click-through rate (CTR) is a prominent metric to boost the popularity of newly published videos on streaming platforms.

This research was supported by the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program IITP-2021-2017-0-01630 supervised by the Institute for Information & communications Technology Promotion (IITP).

978-1-6654-2413-4/21/$31.00 ©2021 IEEE

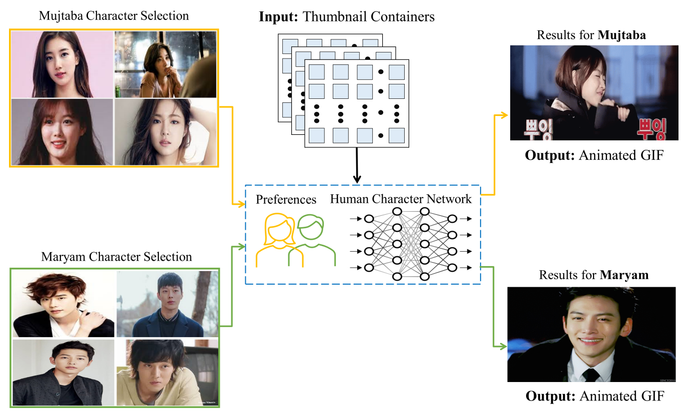

Fig. 1: Popularity and choice of video content often curved on the human character present in the GIF. This paper presents a gender-based human character-oriented animated GIF generation method.

A recent study showed that humans and human-like characters play a significant role in GIF selection and improving the CTR of the video [2]. The popularity of human character-oriented animated GIFs can be anticipated from a GIF search engine named GIPHY; most GIFs contain human/cartoon faces [3]. In most cases, popularity and choice of video content often curved on the human character present in the GIF [4]. The importance of lightweight GIF generation techniques for streaming platforms is highlighted in another study [5]. Motivated by these studies [2–6], we explore a lightweight client-driven technique to generate human-centered animated GIFs for streaming platforms.

This paper proposes a new method to generate animated GIFs according to human gender character-oriented for a video using end-user device computational capabilities. Gender-based human characters-oriented animated GIFs are anticipated to efficiently meet the user's preferences by benefiting from relationships that viewers empathize with a particular character. Figure 1 depicts the conceptual diagram of proposed method. Instead of analyzing full video, the proposed method analyzes the lightweight thumbnail containers to generate human character-oriented animated GIFs. Additionally, it uses

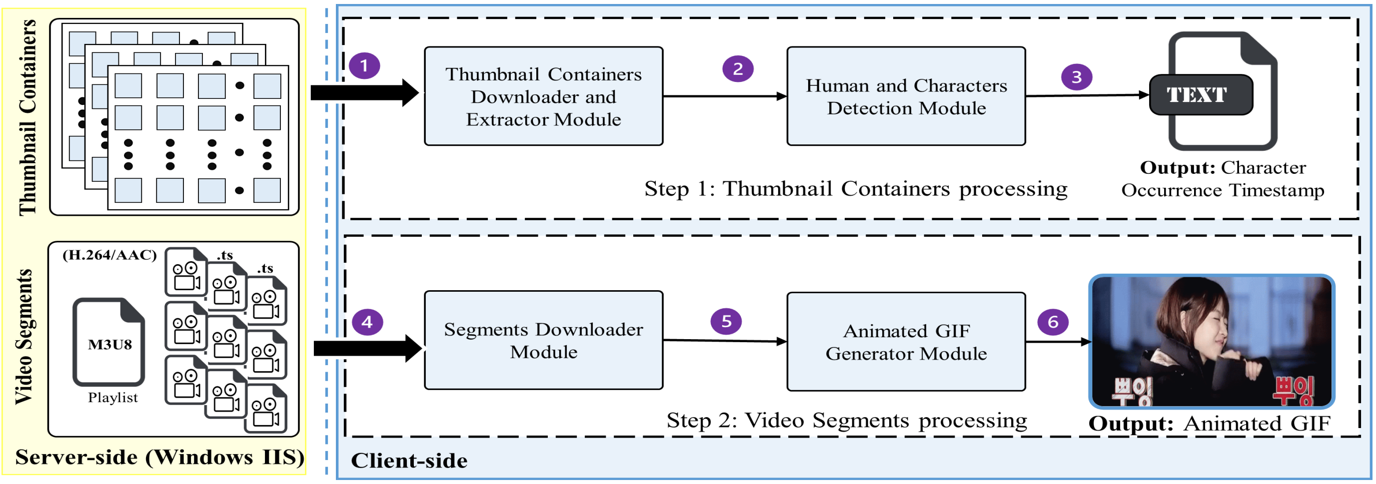

Fig. 2: High-level system architecture of proposed human character-oriented animated GIF generation method.

small segments to generate GIFs, decreasing computational complexity, network bandwidth, and storage demands. To evaluate our method, we have specifically investigated ten music videos that have the story plot. Our proposed method includes:

- A lightweight client-driven framework is designed which analyzes thumbnail containers of the corresponding video in the GIF generation process.
- The thumbnail container analyzer module is designed that consists of fine-tuned InceptionV3 convolutional neural network (CNN) model trained with the annotated CelebA dataset [7]. We use this module to analyze human or human-like characters and their genders from lightweight thumbnail containers.
- Full ten videos are analyzed to demonstrate the feasibility of the proposed method.

The rest of this paper is organized as follows. Section II briefly describes the background of animated GIF images and video analysis. Section III explains the proposed human character-oriented GIF generation method. Performance evaluation and discussion of the proposed GIF generation method are described in Section IV.

II. Related Work

This paper is focus animated GIF images and video analysis. The most recent research works associated with these topics are reviewed in the following section.

A. GIFs Analysis

With the increasing use of social network websites in recent years, animated GIFs are becoming ubiquitous for netizens. A survey of over 3.9 million posts on Tumblr conducted in [1] showed animated GIFs are more attractive than other media types. There is growing interest in research into lightweight GIF generation processes, especially for streaming platforms. Despite the widespread use of GIFs on streaming platforms, little work is performed. In a recent approach[5], researchers proposed the lightweight GIF generation technique for streaming platforms. Instead of analyzing the entire video data, the researchers analyzed the acoustic feature from the climax part of a video in the GIF generation process. This has reduced overall processing time and computational resource demand. The human-centric GIF generation method is proposed by recognizing emotions from the unique properties of GIFs [4]. Similarly, in [8] researchers trained a model for predicting the perceptual sentiment of the user towards animated GIFs. Despite their involvement, it is discovered that viewers may have diverse interpretations of animated GIFs used in communication.

B. Video Analysis

Video analysis is one of the most prominent fields in computer vision research. The CNNs have received a great deal of attention among researchers due to the creation of large datasets such as ImageNet [9, 10]. This advances the development of deep convolutional neural networks to properly perform a variety of machine learning tasks like facilitating movie trailers and GIF generation processes [11, 12]. Various methods have been used for video analysis, such as two-stream networks [13], 3D CNNs [14], and RNNs [15]. Another popular method uses a two-stream to extend 3D CNN's [9]. It is obtained by pretraining a 2D CNN model using the ImageNet [10] dataset and extending the 2D CNN model to a 3D CNN by repeated weighting in a depth-wise manner. These features are locally descriptors obtained using the bag-of-words method or global descriptors retrieved utilizing CNNs.

Lightweight client-driven techniques for GIF generation are still in their infancy. More effective methods are needed to bridge the semantic gap between video understanding and the GIF generation process. Most modern client devices have limited computing power, and using the current approaches will take significant time to inspect full video and create animated GIFs. This paper analyzes lightweight thumbnail containers and proposes effective GIF generation that enhances computational, communication, and storage efficiency. The following section describes the main modules of the proposed method and the GIF generation process.

III. Methodology

This section explains the proposed client-driven animated GIF generation framework. The proposed method consists of the server and client sides. We primarily focus on the client-side implementation. The server-side is configured on Internet Information Services (IIS) running on the Windows 10 operating system [16]. IIS configuration and thumbnail container and video orientation are maintained according to the previous approach [11]. Meanwhile, the client-side is configured on the open-source Ubuntu 18.04 LTS operating system. The client-side is split into two separate components, and each component handles different input data. The high-level architecture of the proposed method is depicted in Figure 2. In the first component, a text-based file is generated by analyzing the transmitted thumbnail containers. The text-based file comprises timestamp information of human character occurrence according to their genders in thumbnails. Thumbnail containers are analyzed using a fine-tuned 2D CNN model trained on the CelebA dataset [7]. The model detects humans or human-like characters along with gender from thumbnails.

Finally, the proposed method uses the character appearance timestamp information to get the video segment and generate an animated GIF in the second component. Instead of analyzing complete video, the proposed approach uses video segments that reduce significant computational complexity, bandwidth, and storage demands. The proposed method can generate animated GIFs according to the gender preferences of the user. The following section provides the details of each part of the proposed method.

A. Thumbnail Containers Analysis

This section provides details corresponding video thumbnail container process. This process is illustrated based on ten videos acquired from the YouTube streaming platform. The details of video titles are shown in Table I. The number of views is collected in March 2021 from the YouTube streaming platform. The resolution of all the videos used in the experiments is 640×480 pixels. We have used the same server-side configuration and thumbnail container generation process as described in the previous approach [11]. Initially, the proposed method downloads thumbnail containers of the corresponding video from the server using the HTTP persistent connection. The transmitted thumbnail containers cover the entire length of the video. The thumbnail container extraction process is explained in the following section.

1) Thumbnail Containers Extractor Module: The proposed method extracts thumbnails using canvas from downloaded thumbnail containers. Later, the extracted thumbnails are analyzed using the 2D CNN model trained on the CelebA dataset [7]. The number and size of thumbnail containers are significantly less than the number of frames in the video. Therefore, the bit rate required during transmission and the demand for computational resources to process is significantly lower. The proposed method uses a canvas to capture all thumbnails separately from the transmitted thumbnail containers. One thumbnail container contains 25 thumbnails, and each thumbnail size is 160×90 (i.e., $width \times height$) pixels. The 2D CNN model is used to analyze the human characters according to their gender from extracted thumbnails. More information on the 2D CNN model is provided in the next section.

2) Human Character Detection Module: The character detection module aims to determine the human character gender from the thumbnails and generate a character occurrence timestamp according to gender in the video. The 2D CNN model is designed for this purpose to examine thumbnails trained on the CelebA dataset [7]. The dataset contains over 200K celebrity facial images and each with 40 attribute annotations. Here, the gender annotations of the images are used in the training process. The InceptionV3 image annotation model that pre-trained on the ImageNet dataset is used to extract features from the image [10, 17]. The classifier layers are replaced with different ones and, all initial layers are locked to avoid retaining. Figure 3 depicts the architecture of the proposed 2D CNN model. Aligned and cropped face images of the CelebA dataset and default training, testing, and validation partitions set of the dataset in the training process.

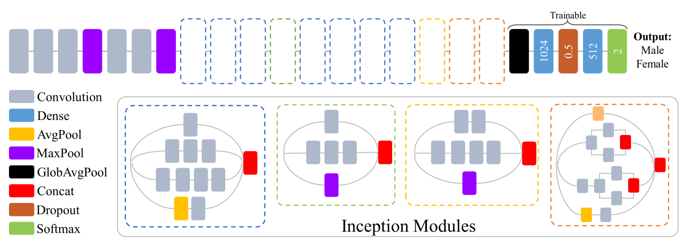

Fig. 3: The architecture of InceptionV3 model.

The data augmentation is used in the training process to reduce overfitting. All images are preprocessed by first cropping the central area and then resizing them to 218×178 pixels before being served as input to the network. Random rotations and shear transformations at 30° and 20° angles, horizontal and vertical shift of 0.2, and random horizontal inversion of the image are performed, respectively. The variant of stochastic gradient descent (SGD) optimization algorithm with default decay (SGDW), the learning rate of 0.01, and momentum of 0.9 are used to train the network [18]. An early stop mechanism with the patience of 10 is used in the experiment. The data is provided in mini-batch with a size of 32 and a learning rate of 0.001 to minimize costs. One thousand epochs are performed to train the sequence pattern of the data. The GeForce RTX 2080 Ti GPU and Keras toolbox are used for deep feature extraction. Section IV-C provides a detailed accuracy analysis of the proposed human character detection model.

The trained network is used to recognize human characters and their gender from the extracted thumbnails. Two separate text-based files are generated, according to the analyzed gender from the thumbnails. The 90% threshold is fixed to maintain

TABLE I: The list of the selected video title and their details used in the experimental evaluation.

S/N	Title	Playtime	FPS	#Frames	#Thumbnail Containers	#Thumbnails	Views	YouTube ID
1	Bhit ja Bhittai	4min 29sec	25	6,740	11	269	5,196,611	AVSPIv6maDU
2	Jind Jaan	3min 41sec	25	5,548	9	221	860,546	WKlAAnj2_uU
3	LeeSSang- Clowns	3min 57sec	30	7,119	10	237	4,023,882	ib-o3OZfqy4
4	Main Pakistan Hoon	3min 33sec	24	5,126	9	213	453,902	IPImomx5I3Q
5	Pehanjo Polly Phenjo Sindh	2min 23sec	25	3,599	6	144	1,422,956	RsNn3qQc_uY
6	Tera Mukhra Haseen	4min 04sec	30	7,299	10	244	4,646,700	nOvWdm9yxVE
7	The Sibbi Song	2min 37sec	24	3,778	7	157	7,416,767	755csdRJ8jg
8	Tum he Sa Ay Mujahido	4min 49sec	25	7,242	12	289	1,960,918	dXiPoIwR93I
9	Wathi Har Har Janam Warbo	6min 48sec	25	10,224	17	409	130,569	J2_8nL4gxdY
10	When You Say Nothing at All	4min 23sec	25	6,575	11	263	44,018,424	IobNcpiwpSc

the quality of the generated GIF. When the accuracy reaches the threshold, all detected thumbnails are included in the text file. Each text-based file is ranked in chronological order based on the detected thumbnails. The generated text-based file is used for analysis to get a specific segment number from the server. Figure 4 shows all the steps that are required in thumbnail containers analysis. The following section provides details of the GIF generation process from text-based files.

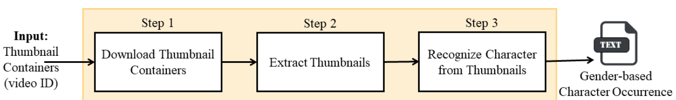

Fig. 4: All the processing steps are required to generate text-based files from thumbnail containers.

B. Video Segment Analysis

The video segment analysis module is designed to examine the segment number from the text-based files generated from the detected thumbnails. Later, we use that information to obtain the corresponding segment from the server and generate an animated GIF. Equation 1 is used to obtain the specific segment number Sn.

$$Sn = \frac{(TC \times 25) + T}{10} \tag{1}$$

In equation 1, TC represents the thumbnail container and T represents the thumbnail number. We multiply because each TC has 25 thumbnails. Since each segment has a 10-second playtime, so we divided it by 10 to get the segment number. HTTP persistent connection is used to obtain all the segments from the server. FFmpeg is used in the proposed method to create GIFs from the segment [19]. The first three seconds of each video segment is used to generate animated GIF in the proposed approach. However, it is extendable to generate a GIF with a specific length. The following section provides details of the performance results of the proposed method.

IV. Results and Discussion

This section presents the experimental evaluation of the baseline and proposed methods. First, the details of the hardware device configuration used in the experimental evaluation are explained. Next, baseline methods are explained. The accuracy of the proposed human recognition model is explained by comparing other methods. Finally, the performance of the proposed method is compared to the baseline method.

A. Hardware Configuration

The client and server hardware devices are configured locally for baseline and proposed methods comparison. The client device is configured on Ubuntu 18.04 LTS that is the open-source operating system. Meanwhile, the server is configured on IIS running on the Windows 10 operating system. All hardware devices are connected locally to Sungkyunkwan university network. Table II depicts the specifications of the hardware devices adopted in the experimental evaluations. The baseline methods are described in the following section.

TABLE II: Specifications of client and server hardware devices used in the experimental evaluation.

Device	CPU	GPU	RAM
Server	Intel Core i7-8700K	GeForce GTX 1080	32 GB
Client	Quad-core 2.10 GHz Xeon	GeForce RTX 2080 Ti	62 GB

B. Baseline Methods

Here, some of the prominent baseline client-drive GIF generation methods for streaming platforms are explained. Those are used to compare the proposed approach. The baseline approaches are as follows:

- **AV-GIF** [5]: It analyzes the entire audio and video files of the video to create animated GIFs. This is the baseline approach that is adopted. To create a GIF, the default parameters are used as described by the authors.
- **Climax-GIF** [5]: It uses acoustic features to analyze the climax portion of the audio and employs segments to generate GIFs. It is the current SoA client-driven animated GIF generation method. Default parameters are applied to generate animated GIFs.

C. Experimental Evaluation Character Detection Model

This section presents the details of the proposed human characters' gender detection model trained on the CelebA dataset. We trained the proposed 2D CNN model with two different optimizer algorithms. First, we train the model on the SGD optimizer algorithm with 0.0001 learning rate and 0.9 momentum. The best validation accuracy is 94.75% of the 2D CNN model is achieved on 50 epochs. Meanwhile, the

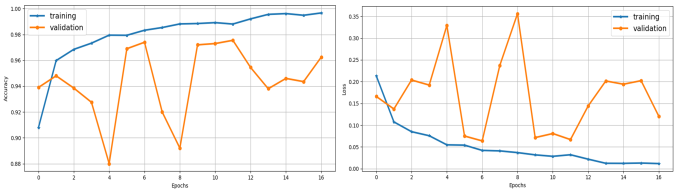

Fig. 5: Plots for both training and testing data are depicts in for accuracy (left) and in for loss during 6 epoch (right).

proposed model using SGDW optimizer algorithm achieved the best validation accuracy that is 97.4% on 6 epochs. The SDGW optimizer algorithm achieved 2.65% better validation accuracy along with reducing the significant processing time. Figure 5 depicts accuracy and loss on the training and testing data of the CelebA dataset.

D. Performance Evaluation of the Proposed Method

This section provides the details of the performance evaluation of the baseline and proposed methods. The ten videos are selected and used for performance evaluation. The computation time of the proposed approach is determined by considering the (i) download thumbnail containers, (ii) obtain thumbnails from thumbnail containers, (iii) recognized human characters according to their genders, and (iv) estimate Sn, download them and finally generate GIFs. All these thumbnails are elected with an accuracy above the threshold of 90%. This threshold is fixed to maintain the quality of the GIF.

Here, we compare the computational time required using the proposed and the baseline approaches to generate an animated GIF. The proposed and baseline methods are configured locally on the client device. Table III shows the computation time required in second for every step using the proposed method to generate GIF. Similarly, Table IV shows the overall computation time required in seconds using the proposed and baseline methods to create the GIF. The overall computation time is much lower than the baseline method. The main reason for the lower computation time of the proposed method used thumbnail containers in the process. Meanwhile, the AV-GIF [5] uses entire video and audio clips, and Climax-GIF [5] uses segments and climax portions of the audio to generate animated GIFs. The thumbnail image is much smaller than the audio, so the proposed method has reduced the computational time in the GIF generation process.

From the perspective of communication and storage, the proposed approach is more effective than the baseline methods. AV-GIF and Climax-GIF [5] baseline methods require entire video and audio to download, respectively. Meanwhile, the proposed approach needs lightweight thumbnail containers to generate animated GIFs. For example, the size of video and audio of The Sibi Song is 11.4 MB and 3.6 MB, respectively. Meanwhile, the size of thumbnail containers for the same video is 468 KB. Thus, the proposed method requires significantly lower communication and storage demanded compared to baseline methods.

TABLE III: The processing time needed in seconds for every step using the proposed method.

S.N	Download TC	Extract T	Recognize Characters	Generate GIF	Total
1	0.32	0.37	8.11	1.42	10.22
2	0.33	0.32	4.4	2.08	7.13
3	0.26	0.39	5.15	1.22	7.02
4	0.31	0.3	4.48	1.79	6.88
5	0.19	0.23	3.14	1.57	5.13
6	0.25	0.29	4.79	1.68	7.01
7	0.26	0.27	3.69	2.06	6.28
8	0.36	0.44	5.27	1.7	7.77
9	0.56	0.6	7.88	1.8	10.84
10	0.3	0.4	5.36	1.65	7.71

TABLE IV: The processing time needed in seconds to generate GIF using baseline and proposed methods.

S.N	AV-GIF [5]	Climax-GIF [5]	Proposed
1	47.66	23.89	**10.22**
2	36.47	16.41	**7.13**
3	39.86	16.67	**7.02**
4	36.97	15.11	**6.88**
5	26.01	11.01	**5.13**
6	39.89	16.95	**7.01**
7	27.21	12.49	**6.28**
8	47.16	19.96	**7.77**
9	66.27	27.71	**10.84**
10	43.02	17.95	**7.71**

The total running duration of the ten videos is 40.73 minutes. AV-GIF [5] requires 410.5 seconds and Climax-GIF [5] requires 178.15 seconds to generate GIFs for these 10 corresponding videos. Meanwhile, the proposed method requires 75.99 seconds to generate GIFs. In this context, the proposed method is 5.4 times more computationally efficient than AV-GIF [5] and 2.34 than Climax-GIF [5] baseline method. These results indicate that the proposed method is significantly computationally effective than baseline methods while using the same computational end-user device.

E. Discussion

The overall effectiveness is evaluated by comparing the baseline with the proposed method. The proposed method achieved significantly higher performance and shorter computation times on end-user devices with the same configuration. Since the proposed method analyzed lightweight thumbnail containers in the GIF generation process, it is 5.4 times more computationally efficient than AV-GIF [5] and 2.34 times than the Climax-GIF [5] baseline method. The proposed method has reduced the overall computational, processing time, communication, and storage requirements of the GIF generation process on the client device.

The proposed system is designed to maintain a wide range of client devices with various computational resource capabilities. Thanks to the simplicity and scalability of implementing multiple device configurations, it can be easily adapted to other animated image formats such as WebP, recommended methods, and streaming protocols like Dynamic Adaptive Streaming over HTTP (DASH). In addition, reducing the

Fig. 6: Sample frames from generated GIF using proposed methods.

computational demand of the server, the proposed method can be extended to protects privacy by utilizing an effective encryption method [20] and three-screen TV solutions [21, 22]. Client-based GIF generation technology is in its infancy, and new scenarios and techniques are open for academic research to explore.

V. Conclusion

This paper proposes the human character-oriented GIF generation technique that leverages the computing power of end-user devices. The proposed method analyzes lightweight thumbnail containers in the GIF generation process. The human characters detected according to their gender using the proposed 2D deep neural network trained on the CelebA dataset. Extensive experiments conducted on ten full videos showed that the proposed method is 2.34 times more computationally efficient than the SoA method on the same configured end-user device. The use of lightweight thumbnail containers enabled the proposed approach to significantly decrease the overall computational, processing time, communication, and storage requirements in the GIF generation process.

References

[1] S. Bakhshi, D. A. Shamma, L. Kennedy, Y. Song, P. De Juan, and J. Kaye, "Fast, cheap, and good: Why animated gifs engage us," in *Proceedings of the 2016 chi conference on human factors in computing systems*, New York, NY, USA, 2016, pp. 575–586.

[2] X. Bost, S. Gueye, V. Labatut, M. Larson, G. Linarès, D. Malinas, and R. Roth, "Remembering winter was coming," *Multimedia Tools and Applications*, vol. 78, no. 24, pp. 35 373–35 399, 2019.

[3] Z. Yang, Y. Zhang, and J. Luo, "Human-centered emotion recognition in animated gifs," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 2019, pp. 1090–1095.

[4] T. Liu, J. Wan, X. Dai, F. Liu, Q. You, and J. Luo, "Sentiment recognition for short annotated gifs using visual-textual fusion," *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 1098–1110, 2020.

[5] G. Mujtaba, S. Lee, J. Kim, and E.-S. Ryu, "Client-driven animated gif generation framework using an acoustic feature," *Multimedia Tools and Applications*, 2021.

[6] S. Sharif, G. Mujtaba, and S. Uddin, "Edgenet: A novel approach for arabic numeral classification," *arXiv preprint arXiv:1908.02254*, 2019.

[7] Z. Liu, P. Luo, X. Wang, and X. Tang, "Large-scale celebfaces attributes (celeba) dataset," *Retrieved August*, vol. 15, no. 2018, p. 11, 2018.

[8] W. Chen, O. O. Rudovic, and R. W. Picard, "Gifgif+: Collecting emotional animated gifs with clustered multi-task learning," in *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2017, pp. 510–517.

[9] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[11] G. Mujtaba and E.-S. Ryu, "Client-driven personalized trailer framework using thumbnail containers," *IEEE Access*, vol. 8, pp. 60 417–60 427, 2020.

[12] G. Mujtaba, S. Kim, E. Park, S. Kim, J. Ryu, and E.-S. Ryu, "Client-driven animated keyframe generation system using music analysis," in *Proceedings of the Korean Society of Broadcast Engineers Conference*, Jeju, Korea, 2019, pp. 173–175.

[13] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in neural information processing systems*, vol. 27, pp. 568–576, 2014.

[14] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[15] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[16] M. O Leary, "Iis iis iis and modsecurity," in *Cyber Operations*. Springer, 2019, pp. 789–819.

[17] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[18] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017. [Online]. Available: http://arxiv.org/abs/1711.05101

[19] FFmpeg. (2020) Ffmpeg github page. [Online]. Available: https://github.com/FFmpeg/FFmpeg

[20] G. Mujtaba, M. Tahir, and M. H. Soomro, "Energy efficient data encryption techniques in smartphones," *Wireless Personal Communications*, vol. 106, no. 4, pp. 2023–2035, 2019.

[21] H.-W. Kim, T. T. Le, and E.-S. Ryu, "360-degree video offloading using millimeter-wave communication for cyberphysical system," *Transactions on Emerging Telecommunications Technologies*, vol. 30, no. 4, p. e3506, 2019.

[22] J.-B. Jeong, S. Lee, D. Jang, and E.-S. Ryu, "Towards 3dof+ 360 video streaming system for immersive media," *IEEE Access*, vol. 7, pp. 136 399–136 408, 2019.